

1. Introduction aux Systèmes d'exploitation

ASR1 - Systèmes d'exploitation

Semestre 1, année 2009-2010

Département d'informatique
IUT Bordeaux 1

Septembre 2009



Contenu

- ▶ à quoi ça sert
- ▶ grandes fonctions
- ▶ systèmes existants
- ▶ ce que vous allez utiliser

A quoi ça sert ?

Un système d'exploitation sert à **exploiter** une machine

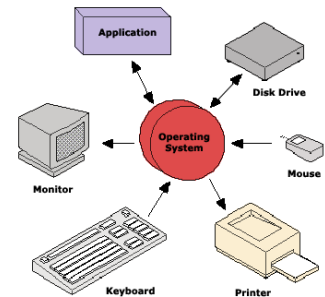


(Operating System)

Définition

Système d'exploitation :
ensemble de programmes
qui assurent la liaison
entre

- ▶ les ressources matérielles
- ▶ les applications



Fonctions d'un système d'exploitation

Dirige le fonctionnement de l'ordinateur

- ▶ initialise le matériel
- ▶ fonctions de base pour piloter les périphériques
- ▶ gestion des fichiers
- ▶ assure la gestion, l'ordonnancement et la communication des processus (tâches)
- ▶ ...

On en trouve partout



Types de systèmes

Classification

- ▶ **Spécialisés** pour un matériel : (système d'alarme, imprimante multi-fonction, console de jeux, magnétoscope ...)
- ▶ **Généralistes** : permettent de faire tourner des programmes très divers sur des matériels différents.

Autres critères Mono/multi-tâche, postes de travail/serveur, embarqués/mono/multi-utilisateurs...

Le marché

Il existe des centaines de systèmes :

- ▶ Windows,
- ▶ Unix : Linux, Solaris, AIX, HP-UX, FreeBSD ...
- ▶ GCOS, VMS, AS400 ...
- ▶ ...

Inconnus du grand public :

- ▶ mainframes
- ▶ systèmes embarqués

- ▶ systèmes d'exploitation embarqués,
- ▶ applications mobiles.
- ▶ virtualisation



Vous utiliserez principalement

- ▶ Linux
- ▶ Windows XP Pro



2. Le cours de systèmes d'exploitation

Objectifs du cours ASR1

Contenu

- ▶ Objectifs du cours
- ▶ Liens avec autres enseignements

- ▶ Notions sur les systèmes d'exploitation
- ▶ Connaître l'environnement local
- ▶ Utilisation concrète des Systèmes Informatiques (USI)
- ▶ Utilisation du langage de commande
 - ▶ Commandes de base
 - ▶ Filtres
 - ▶ Programmation des scripts

Lien avec autres enseignements

3. Environnement de travail

Les cours de systèmes d'exploitation

- S1 USI + scripts
utiliser/combinaison des commandes
- S2 Fonctionnement des systèmes d'exploitation
structure et organisation interne
- S3 Programmation système et réseau
utiliser les fonctions du système d'exploitation (API) dans les programmes

API : *Application Programming Interface*

L'environnement de travail du département informatique de l'IUT de Bordeaux

Contenu

- ▶ les utilisateurs
- ▶ le réseau du département
- ▶ comment se connecter
- ▶ principales adresses

Les utilisateurs

Le réseau du département (Bordeaux)

- ▶ environ 250 étudiants (11 groupes)
 - ▶ DUT A1 = 100, A2 = 90
 - ▶ DUT AS = 20
 - ▶ LP = 20 + 20 + 15
- ▶ une cinquantaine d'enseignants
 - ▶ permanents
 - ▶ extérieurs
 - ▶ vacataires et professionnels

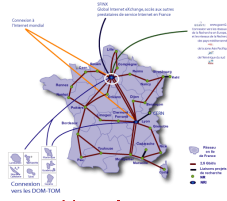
Réseau **filaire** (ethernet) et **Wifi** reliant

- ▶ une centaine de **stations**, dual boot Linux+Windows (salles en libre service)
- ▶ une douzaine de **serveurs** (fichiers partagés, services)
- ▶ des **imprimantes** (une par salle)
- ▶ ...



Le département est relié au reste du monde via

- ▶ **REAUMUR** : REseau Aquitain des Utilisateurs des Milieux Universitaires et de la Recherche
- ▶ **RENATER** : REseau NATIONAL pour l'Enseignement et la Recherche



Utilisation des équipements limitée aux usages pédagogiques (charte REAUMUR).

Chaque utilisateur possède un **compte** protégé par un **mot de passe**
Le compte donne accès

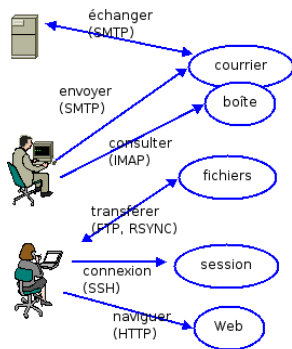
- ▶ à des fichiers personnels
- ▶ à une boîte aux lettres
- ▶ à des services

Un utilisateur est **responsable** de son compte.

Le domaine **info.iut.u-bordeaux1.fr (services)**

Le domaine **info.iut.u-bordeaux1.fr (serveurs)**

Les **services** accessibles depuis l'extérieur



Adresses **publiques** des services accessibles depuis l'extérieur

Service	? .info.iut.u-bordeaux1.fr
serveurs web	www, pedago
connexions à distance (ssh)	e1, e2
boîtes aux lettres	imap
transferts de fichiers	ftp
...	...

Adresses e-mail au département :

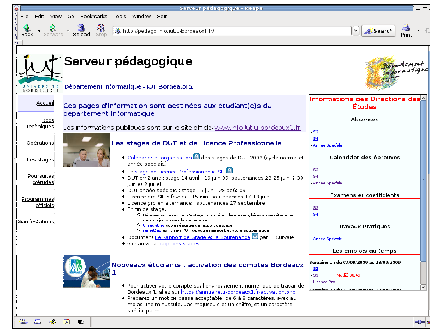
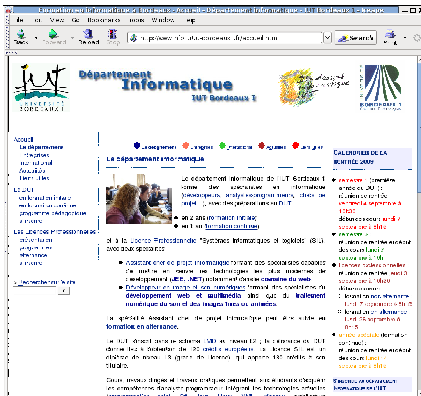
compte@info.iut.u-bordeaux1.fr

Site officiel

Site pédagogique

www.info.iut.u-bordeaux1.fr

pedago.info.iut.u-bordeaux1.fr



(usagers du département)

Le domaine **info.privé**

4. Utiliser Linux

Adresses **privées** du réseau interne

- ▶ stations de travail
- ▶ serveurs à usage interne

Service	? .info.privé
serveurs web	www, pedago
boîtes aux lettres	imap
transferts de fichiers	ftp
serveurs unix	tuba, helicon
serveurs windows	apopsis
...	...

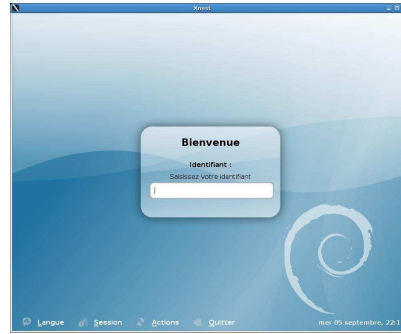
Contenu

- ▶ c'est quoi Linux ?
- ▶ comment se connecter
- ▶ le bureau GNOME

Linux : système d'exploitation

- ▶ créé en 1992 par Linus Torvalds
- ▶ Membre de la famille **Unix** (1969) : système généraliste, multi-tâche, multi-utilisateurs, partages de ressources, réseau. Conforme à la norme **POSIX**.
- ▶ En réalité Linux est le nom d'un **noyau** utilisé dans de nombreuses **distributions** : Ubuntu, Mandriva, Debian, Gentoo, Slackware, etc.

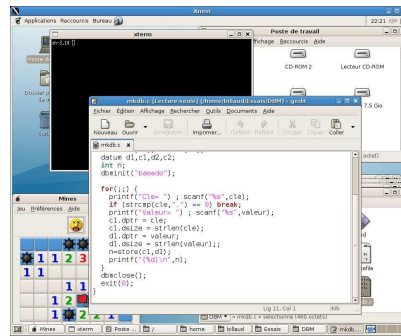
Ici, distribution Debian 5.0 "Lenny"



Le bureau GNOME (vide)



Fenêtres



Interaction graphique

En mode graphique, interaction basée sur

- ▶ menus, sous-menus
- ▶ fenêtres, icônes, barre de titres
- ▶ pointeurs, curseurs
- ▶ souris (clics, déplacements), modificateurs
- ▶ clavier
- ▶ ...

5. Lignes de commandes

Contenu

- ▶ principe
- ▶ syntaxe
- ▶ exemple
- ▶ où trouver la documentation ?

Interface lignes de commandes

Les "Interprètes de ligne de commande" (*shells*) permettent de lancer des actions décrites par des lignes de commandes.

Exemples :

- ▶ `ls`
- ▶ `cp /mnt/usbdisk/*.mp3 Ma_Musique`
- ▶ `ls Ma_Musique | mpage -4 -Pbureau421`

Interface graphique ou texte ?

- ▶ interfaces graphiques : plus accessibles aux utilisateurs *occasionnels*
- ▶ ligne de commande : **usage plus efficace**, nécessitant un investissement (rentable) pour les utilisateurs professionnels. Permet à la fois
 - ▶ usage interactif
 - ▶ exécution de **scripts** (suites de commandes enregistrées), automatisation de tâches répétitives.

Remarque : complémentarité : un script peut être lancé par une interface graphique, et inversement.

Une commande comporte

- ▶ le nom d'un programme
- ▶ des options (précédées par un tiret)
- ▶ des paramètres

Ces éléments sont séparés par des espaces.

Exemple

```
a2ps -r -Pbureau421 premier.cc doc.txt
```

Exemple

```
a2ps -r -Pbureau421 premier.cc doc.txt
```

Détails expliqués

- ▶ a2ps : programme de formatage et impression
- ▶ -r : orientation "paysage"
- ▶ -Pbureau421 : sur l'imprimante bureau421
- ▶ premier.cc doc.txt : noms des fichiers à imprimer

Où trouver de la documentation

Documentation intégrée (option -h ou --help)

```
$ cp --help
Usage: cp [OPTION]... [-T] SOURCE CIBLE
ou: cp [OPTION]... SOURCE... RÉPERTOIRE
ou: cp [OPTION]... --target-directory=RÉPERTOIRE SOURCE...
Copier la SOURCE vers la DESTINATION, ou de multiples SOURCES
vers un RÉPERTOIRE.

Les arguments obligatoires pour les options de formes longues
le sont aussi pour les options de formes courtes.
-a, --archive          identique à -dpr
--backup[=CONTROL]   archiver chaque fichier de destination
                      identique à --backup mais sans argument
-b                    copier le contenu des fichiers spéciaux
                      en mode récursif
--copy-contents
-d                    identique à --no-dereference --preserve=link
...
```

Où trouver de la documentation (suite)

Pages de manuel (commande man)

```
$ man cp
CP(1)                                User Commands                                CP(1)
NAME
  cp - copy files and directories
SYNOPSIS
  cp [OPTION]... [-T] SOURCE DEST
  cp [OPTION]... SOURCE... DIRECTORY
  cp [OPTION]... -t DIRECTORY SOURCE...
DESCRIPTION
  Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.
  Mandatory arguments to long options are mandatory for short options
  too.
-a, --archive          same as -dpr
--backup[=CONTROL]   make a backup of each existing destination file
```

Où trouver de la documentation (suite 2)

Autres sources de documentation

- ▶ commande info
- ▶ documentations installées (/usr/share/doc)
- ▶ sites pédagogiques
- ▶ forums d'aide
- ▶ moteurs de recherche
- ▶ êtres humains
- ▶ ...

6. Arborescence

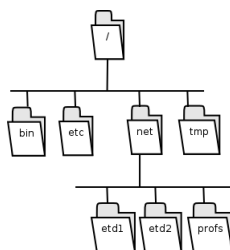
Contenu

- ▶ arborescence = fichiers + répertoires
- ▶ chemins d'accès relatifs et absolus
- ▶ répertoire de travail

Arborescence des fichiers et répertoires

Les données et les programmes sont stockés dans une arborescence de fichiers et répertoires.

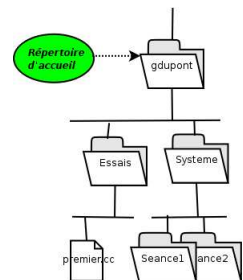
/	"racine"
/tmp	fichiers temporaires
/bin	commandes
/etc	configuration système
...	...
/net	fichiers partagés
/net/etd1	étudiants 1ère année
/net/profs	profs
...	...



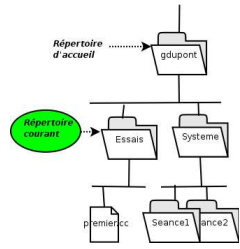
Répertoire d'accueil

Le répertoire d'accueil d'un utilisateur

- ▶ hébergé sur un serveur commun
- ▶ partagé entre toutes les machines du département
- ▶ protégé par des droits d'accès



- ▶ **Chemin d'accès** : désignation de la position d'un fichier (ou d'un répertoire) dans l'arborescence
- ▶ **Chemin relatif** : en fonction du répertoire courant
- ▶ Exemples :
 - ▶ premier.cc,
 - ▶ ..
 - ▶ ../Systeme/Seance2



Chemin d'accès absolu : indépendant du répertoire courant.
Exemples

- ▶ /net/etd1/gdupont/Essais/premier.cc
- ▶ gdupont/Essais/premier.cc
- ▶ ~/Essais/premier.cc

Exemple

7. Commandes utiles

La commande

```
cp /mnt/usbdisk/*.mp3 Ma_Musique
```

copie des fichiers (commande cp = copy)

Quoi? Tous les fichiers

- ▶ du répertoire /mnt/usbdisk (chemin absolu)
- ▶ dont le nom se termine par .mp3

Où? Dans le sous-répertoire Ma_Musique du répertoire courant (chemin relatif)

Contenu

- ▶ créer/supprimer/déplacer des fichiers, des répertoires
- ▶ développer des programmes
- ▶ éditeur emacs

Premières commandes

Objectif à court terme

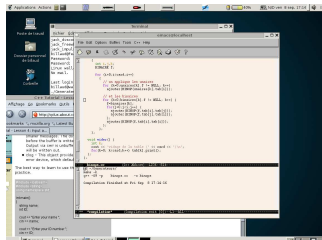
mkdir	répertoire	crée (make) un répertoire (directory)
rmdir	répertoire	supprime (remove) un répertoire
cd	répertoire	change répertoire courant
pwd		affiche (print) chemin du rép. courant
ls		affiche (list) contenu du rép. courant
cp	fichier... destination	copie (copy) fichiers
mv	fichier... destination	déplace (move) fichiers
rm	fichier...	supprime fichiers
cp -r	répertoire... destination	copie récursive répertoire
rm -r	répertoire...	suppression récursive répertoire

- ▶ commandes de base
 - ▶ ...
- ▶ développement de programmes
 - ▶ édition de texte : pour taper le **texte source**, le modifier etc.
 - ▶ compilation : traduire le **source** en **binnaire exécutable**
 - ▶ exécution

Commandes nécessaires

IDE

- ▶ Éditeur de textes : **emacs** `essai.cc`



- ▶ Compilateur `g++ -o essai essai.cc`
- ▶ Exécution : `essai`

Les environnements de développement intégrés (IDE) permettent d'enchaîner rapidement ces activités : **Emacs**, **Eclipse**, ...

